Last modification: 18/02/2021

Version: 1.0

IWAST Consumption

Current consumption measurements

Author(s):

Pierre VERHULST

Table of contents

1. The Motherboard

1.1 Description

The motherboard represents the main part of the IWAST embedded system. It's a necessary component used to connect all sensors and to collect all their data. These ones are then sent to TTN[1] by a LoRa chip. This board is based on a microcontroller ARM® Cortex®-M0+ which can manage all features of the system. It's also the bridge between the computer and the IWAST system used to set up the IWAST configurations. So, we can configure the system via the QT-config app if we connect a USB wire between a computer and the motherboard.

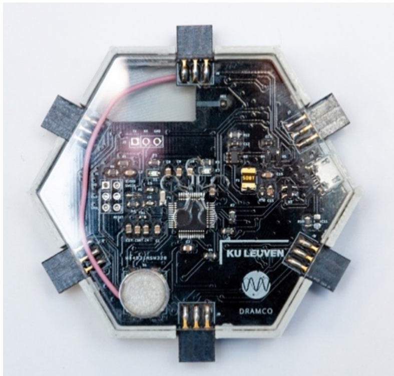

Figure 1 The Motherboard

We can list their main features:

- We can connect up to six sensors boards and collect data from them.
- We can set up several kinds of configurations for the connected sensor boards (polling and threshold).
- The motherboard sends a status message a few times after starting up.
- The only button present is used to restart the motherboard and to set a new configuration via the IWAST Configurator.

1.2 Hardware analysis

<u>ATSAMD21G18 - Main MCU</u>

This board is built around a 32-bit ATSAMD21G18 microcontroller (48 MHz). Its wiring is classic and respects all specifications from the Datasheet. The power supply is set to 3.3 V and all decoupling capacitors respect the recommended values of the Datasheet.

We can quote the two pins PA23 and PA22 used to establish the I^{2}C communications between this MCU (the master) and all the connected sensor boards (the slaves). This bus is built with pull-up resistors.

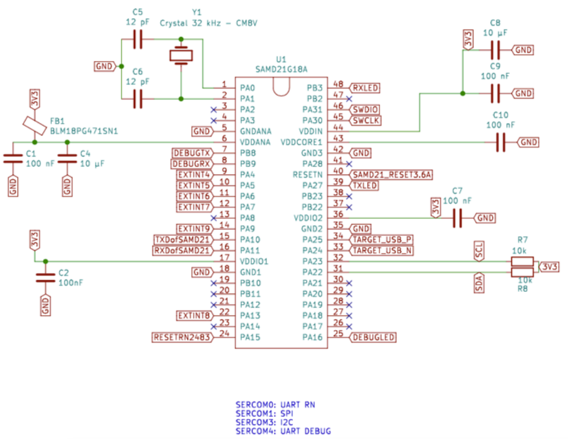

Figure 2 Schematic of the MCU's Motherboard

[1] TTN = *The Thing Network*

This MCU allows us to select specific modes in order to control the power consumption of the chip. There is a normal mode (not efficient in terms of power consumption) and a sleep mode. With this chip, two possibilities are possible for the sleep mode: the IDLE mode and the STANDBY mode. This second option is the best if we want to consume the least amount of power. According to the Datasheet[2] :

Mode	Conditions	T_A	Min.	Typ.	Max.	Units
ACTIVE	CPU running a While(1) algorithm	25°C	3.11	3.37	3.64	mA
		85°C	3.24	3.48	3.76	
	CPU running a While(1) algorithm V_{DDIN}=1.8V, CPU is running on Flash with 3 wait states	25°C	3.10	3.36	3.64	
		85°C	3.24	3.48	3.75	
	CPU running a While(1) algorithm, CPU is running on Flash with 3 wait states with GCLKIN as reference	25°C	60*freq + 74	60*freq + 136	62*freq + 196	µA (with freq in MHz)
		85°C	62*freq + 154	62*freq + 228	62*freq + 302	
	CPU running a Fibonacci algorithm	25°C	4.12	4.53	4.92	mA
		85°C	4.27	4.63	4.98	
	CPU running a Fibonacci algorithm V_{DDIN}=1.8V, CPU is running on flash with 3 wait states	25°C	4.12	4.53	4.92	
		85°C	4.27	4.63	4.98	
	CPU running a Fibonacci algorithm, CPU is running on Flash with 3 wait states with GCLKIN as reference	25°C	86*freq + 76	88*freq + 136	88*freq + 196	µA (with freq in MHz)
		85°C	88*freq + 156	88*freq + 230	88*freq + 302	
	CPU running a CoreMark algorithm	25°C	5.78	6.32	6.80	mA
		85°C	5.93	6.47	7.00	
	CPU running a CoreMark algorithm V_{DDIN}=1.8V, CPU is running on flash with 3 wait states	25°C	5.17	5.60	5.96	
		85°C	5.35	5.73	6.10	
	CPU running a CoreMark algorithm, CPU is running on Flash with 3 wait states with GCLKIN as reference	25°C	106*freq + 78	106*freq + 136	108*freq + 196	µA (with freq in MHz)
		85°C	106*freq + 154	108*freq + 232	108*freq + 310	
IDLE0	Default operating conditions	25°C	1.89	2.04	2.20	mA
		85°C	1.98	2.14	2.33	
IDLE1	Default operating conditions	25°C	1.34	1.46	1.58	
		85°C	1.41	1.55	1.71	
IDLE2	Default operating conditions	25°C	1.07	1.17	1.28	
		85°C	1.13	1.27	1.40	
STANDBY	XOSC32K running RTC running at 1kHz	25°C	-	4.06	12.8	µA
		85°C	-	55.2	100	
	XOSC32K and RTC stopped	25°C	-	2.70	12.2	
		85°C	-	53.3	100	

Table 1 Current consumptions for several modes of ATSAMD21

[2] Table from the Datasheet "*SAM D21/DA1 Family Low-Power, 32-bit Cortex-M0+ MCU with Advanced Analog and PWM*" - p. 869-870

This table is quite interesting for the analysis of the STANDBY mode. Indeed, all peripherals are disabled in this mode (except for the RTC if enabled by the software). It's more complex to analyse the ACTIVE mode because it depends on the program. So, we need to know which peripherals are enabled/disabled in the code for the analysis of the active mode. We can estimate the consumption for each specific peripheral according to the Datasheet[3] data:

Peripheral	Conditions	Typ.	Units
RTC	f_{GCLK_RTC} = 32kHz, 32bit counter mode	7.4	µA
WDT	f_{GCLK_WDT}=32kHz, normal mode with EW	5.5	µA
ACx	Both f_{GCLK}=8MHz, Enable both COMP	31.3	µA
TCC2	f_{GCLK}=8MHz, Enable + COUNTER	95.5	µA
TCC1	f_{GCLK}=8MHz, Enable + COUNTER	167.5	µA
TCC0	f_{GCLK}=8MHz, Enable + COUNTER	180.3	µA
SERCOMx.I2CM[(2)]	f_{GCLK}=8MHz, Enable	69.7	µA
SERCOMx.I2CS	f_{GCLK}=8MHz, Enable	29.2	µA
SERCOMx.SPI	f_{GCLK}=8MHz, Enable	64.6	µA
SERCOMx.USART	f_{GCLK}=8MHz, Enable	65.5	µA
I2S[(3)]	$f_{GCLK\_I2S\_0}$=12.288MHz with source FDPLL with f_{FDPLL}=49,152MHz	26.4	µA
DMAC[(4)]	RAM to RAM transfer	399.5	µA

Table 2 Typical peripheral current consumption

RN2483 - LoRa® chip

This chip is used to transmit data from the main MCU to the TTN. The communication between both is done via UART protocol. The default baud rate is 57600 bps (important to know if we want to estimate the time of a transmission).

The wiring respects all specifications from the Datasheet. We don't need to program this chip. So, the analysis of its current consumption is simpler than the MCU.

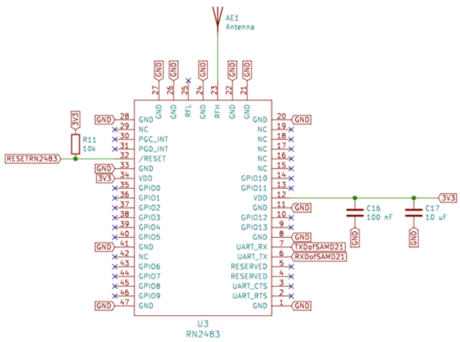

Figure 3 Wiring of the LoRa chip

According to the Datasheet[4] :

Mode	Temperature (°C)	Typical Current (mA)		
		VDD = 2.1V	VDD = 3.3V	VDD = 3.6V
Idle	-40 to +85	1.7	2.8	3.1
Transmit	25	28.6	38.9	44.5
Sleep	-40	0.0011	0.0013	0.0014
	25	0.0015	0.0016	0.0016
	85	0.002	0.0026	0.0026
Receive	-40 to +85	12.96	14.22	14.69

Table 3 Current Consumption for the LoRa chip

So, according to the code of the MCU, we can estimate the current consumption of the chip depending on its mode.

[3] Table from the Datasheet "*SAM D21/DA1 Family Low-Power, 32-bit Cortex-M0+ MCU with Advanced Analog and PWM*" – p. 873
[4] Table from the Datasheet "*RN2483 Low-Power Long Range LoRa® Technology Transceiver Module*" - p. 7

LED

The LED used for the debugging and the two LEDs used when we set up the configurations can consume a little power. We must therefore take them into consideration. To estimate their current consumption, we need to know their typical forward voltage.

The LED used for the debugging is a part of the short travel key switch from MARQUARDT®. We use a yellow LED and the Datasheet[5] gives us the forward voltage: 1.8 V. So, with the serial resistor R4 (1 kΩ) and the power supply (3.3 V), it's easy to determine the current that will pass through when the LED is active:

$$I\,[A] = \frac{U\,[V]}{R\,[\Omega]} = \frac{3.3 - 1.8}{1000} = 1.5\,mA$$

The two LEDs used when we set up the configuration also have a typical forward voltage of 1.8 V. So, with the two resistor R2 and R3 (both 470 Ω), it's easy to determine the current that will pass through each LED when they are active:

$$I\,[A] = \frac{U\,[V]}{R\,[\Omega]} = \frac{3.3-1.8}{470} = 3.2\,mA$$

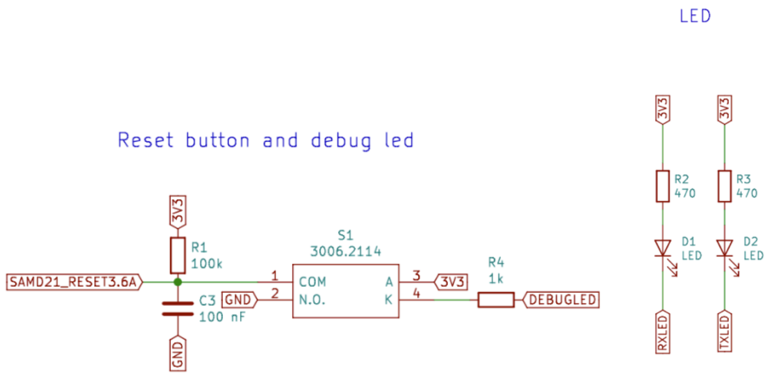

Figure 4 Reset button / Debug LED / LEDs for communications

[5] From the Datasheet "*MARQUARDT® Technical Specification K30062000*" - p.17

Power Management

Two solutions are possible to supply the system with the correct value of 3.3 V. The first is the easiest way. It consists to supply the system with Power module which directly delivers 3.3 V on the VBAT pin. The second solution is to supply the board thanks to the USB connection with a computer. This solution needs a regulator (TPS7A0533) to obtain an accurate value of 3.3 V and this part consumes also some current.

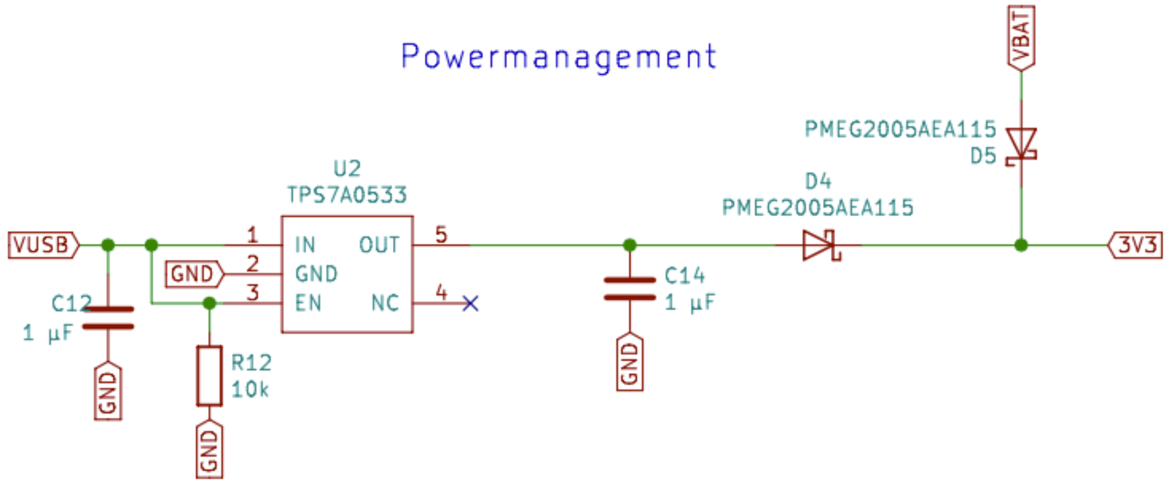

Figure 5 Power management of the motherboard

If we use this regulator, we normally use 10 nA for the enable pin (according to the Datasheet[6])

N.B: *The IWAST system aims to be self-sufficient thanks to the Power module. So, this regulator is useless with the Power module. Its current consumption must not be considered in the calculation.*

[6] From the Datasheet "*TPS7A05 1-µA Ultralow I_Q, 200-mA, Low-Dropout Regulator in a Small-Size Package*" - p.8

1.3 Measurements

Modus operandi

1. Plug the motherboard and the *Otii ARC* according to the scheme fig.6 (refer to annex A in order to understand how to configure the measuring device *Otii ARC*).

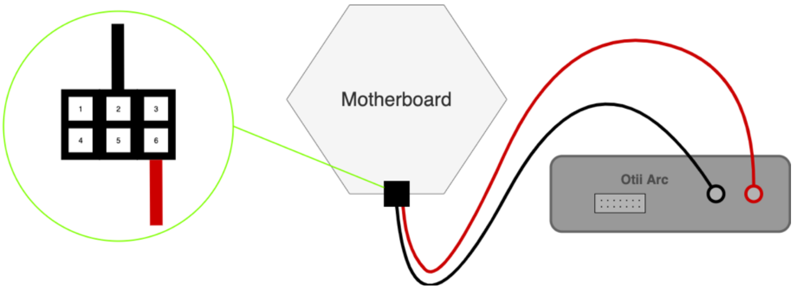

Figure 6 Connect the motherboard and the Otii ARC

2. Set the system configurations via the *IWAST configurator app* (refer to annex B in order to understand how to use this application). Several configurations must be set:

    2.1 **Motherboard without accumulation data (ID 0)**:
    - The accumulation data feature must be disabled

    2.2 **Motherboard with accumulation data (ID 1)**:
    - The accumulation data feature must be enabled

3. Collect the current measurements for each configuration at different moment. The test should last minimum 15 min to be relevant. With this physical configuration, we can collect the values of the current flowing between the power supply and the motherboard (location 1). Multiple scenarios must be analysed with this module:

    A. When nothing happens (static power / sleep mode)
    B. When motherboard wakes up each minute (dynamic power 0)
    C. When data are sent by the motherboard (data sending power → Status Message)

4. Collect measurements for special cases. Multiple scenarios must be analysed:

    Special Meas. 1.1 : ***DATA_TX Time variation depending on the spreading factor (Status Message)***

    This special measurement must be taken to determine if the LoRa airtime estimation (determined by an estimation online tool[7]) is correct or not. So we collect the airtime of the LoRa castle for each possible spreading factor (7 to 12) and we note the difference between the estimation and the reality. This test must be performed with the status message consisting of a 7-byte payload.

[7] Link for the tool: https://www.loratools.nl/#/airtime

<u>Special Meas. 1.2 :</u> ***Average Current variation for six different motherboards (SF = 11 for each motherboards)***

This special measurement must be taken to evaluate the average consumption of the static power for multiple motherboards. In the static power mode, we noticed that the static current can vary greatly depending on several motherboard. So, to have a correct estimation of the static current, we must analyse multiple static current of multiple motherboards under the same conditions.

<u>Warning</u> : If we want to measure the currents of the motherboard and use these ones for the estimation, we must connect a sensor board during the configuration and disconnect it just before doing the measurements. When no sensor boards are connected during the configuration, it seems that motherboard consumes ± 10 uA less. So, to do measurements with motherboard alone, respect this specific modus operandi :

- Connect the motherboard and a sensor board together.
- Use the IWAST Configurator but don't set any kinds of configurations related to the sensor board. Just click on the disconnect button.
- Disconnect the USB cable and the sensor board to have a correct current measurement of the motherboard alone.

Results

Configuration 2.1 : Motherboard without accumulation data (ID 0)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	388.6 µA
Minimum Value	26.74 µA
Average Value	27.94 µA

Remarks :

This average value is correct. However, it must be taken into account that this value can vary greatly depending on the chosen motherboard (probably due to welding). So we must take measurements with multiple motherboards to obtain a correct average value and to determine an error level.

Measure	1	2	3	4	5
Max. Value	387	403	405	408	340
Min. Value	26.9	26.8	26.7	26.7	26.6
Aver. Value	28.1	28	27.9	27.9	27.8

Scenario B – Motherboard wakes up (dynamic power 0)

Location 1	Average
Maximum Value	9154 µA
Minimum Value	-267.8 µA
Average Value	8044 µA
Energy	0.611 µWh
Interval Time	82.88 ms

Remarks :

Measure	1	2	3	4	5
Max. Value	9140	9170	9140	9130	9190
Min. Value	-270	-270	-286	-242	-271
Aver. Value	8040	8060	8050	8040	8030
Energy	0.611	0.611	0.611	0.611	0.611
Interval Time	82.9	82.8	82.8	82.9	83

Scenario C - Data sent by the motherboard (data sending power -> Status Message)

Location 1	Average
WUT_ST Time	82.22 ms
WUT_ST Energy	1.094 µWh
DATA_TX Time	744.62 ms
DATA_TX Aver. Cur	47300 µA
DATA_RX Time	2299.66 ms
DATA_RX Energy	39.06 µWh

Remarks :

We are supposed to have DATA_TX Time = 495.62 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 249 ms

Measure	1	2	3	4	5
WUT_ST Time	82	81.9	82.5	82.6	82.1
WUT_ST Energy	1.09	1.09	1.1	1.1	1.09
DATA_TX Time	745	745.1	744	744	745
DATA_TX Aver. Cur	47100	47400	47200	47300	47500
DATA_RX Time	2299.2	2299.9	2299.7	2299.8	2299.7
DATA_RX Energy	39.1	39	39	39.1	39.1

Configuration 2.2 : Motherboard with accumulation data (ID 1)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	388.6 µA
Minimum Value	26.62 µA
Average Value	27.9 µA

Remarks :

This average value is correct. However, it must be taken into account that this value can vary greatly depending on the chosen motherboard (probably due to welding). So we must take measurements with multiple motherboards to obtain a correct average value and to determine an error level.

Measure	1	2	3	4	5
Max. Value	400	403	403	399	338
Min. Value	26.6	26.6	26.7	26.6	26.6
Aver. Value	28	27.9	27.9	27.9	27.8

Scenario B – Motherboard wakes up (dynamic power 0)

Location 1	Average
Maximum Value	9212 µA
Minimum Value	-252.4 µA
Average Value	8158 µA
Energy	0.6112 µWh
Interval Time	81.76 ms

Remarks :

Measure	1	2	3	4	5
Max. Value	9150	9160	9340	9180	9230
Min. Value	-277	-284	-215	-263	-223
Aver. Value	8140	8160	8190	8140	8160
Energy	0.611	0.611	0.612	0.611	0.611
Interval Time	81.9	81.8	81.5	81.9	81.7

Scenario C - Data sent by the motherboard (data sending power -> Status Message)

Location 1	Average
WUT_ST Time	82.34 ms
WUT_ST Energy	1.098 µWh
DATA_TX Time	745.14 ms
DATA_TX Aver. Cur	47600 µA
DATA_RX Time	2300.24 ms
DATA_RX Energy	39.06 µWh

Remarks :

We are supposed to have DATA_TX Time = 495.62 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 249.52 ms

Measure	1	2	3	4	5
WUT_ST Time	82.5	82.5	82.2	82	82.5
WUT_ST Energy	1.1	1.1	1.1	1.09	1.1
DATA_TX Time	745	745	743.9	744.9	746.9
DATA_TX Aver. Cur	47700	47000	47900	47400	48000
DATA_RX Time	2298.8	2300.6	2302.4	2299.9	2299.5
DATA_RX Energy	39.1	39	39.1	39	39.1

Special Measurements

Special Measurement 1.1 - DATA_TX Time variation depending on the spreading factor (Status Message)

Location 1	Measure	Theory	Delta
DATA_TX Time (SF = 7)	60.23 ms	36.10 ms	24.13 ms
DATA_TX Time (SF = 8)	107.87 ms	72.19 ms	35.68 ms
DATA_TX Time (SF = 9)	189.10 ms	123.9 ms	65.2 ms
DATA_TX Time (SF = 10)	374.13 ms	247.81 ms	126.32 ms
DATA_TX Time (SF = 11)	744.62 ms	495.62 ms	249 ms
DATA_TX Time (SF =12)	1486.86 ms	991.23 ms	495.63 ms

Remarks :

Special Measurement 1.2 - Average Current variation for six different motherboards (SF = 11 for each motherboards)

Location 1	Average
Maximum Value	379.5 µA
Minimum Value	26.53 µA
Average Value	27.89 µA
Extensive	2.52 µA

Remarks :

Measure	Aver. MB 1	Aver. MB 2	Aver. MB 3	Aver. MB 4	Aver. MB 5	Aver. MB 6
Max. Value	401.6	398.6	365	358.4	388.6	364.8
Min. Value	26.44	27.34	25.16	26.56	26.74	26.94
Aver. Value	27.86	28.94	26.42	27.8	27.94	28.36

1.4 Conclusion

At the end of the tests, **no major differences were noticed between ID0 and ID1**. Therefore, we can conclude that the enabling/disabling of the data accumulation feature has no influence on the motherboard consumption. For the total estimation, it's not valuable to keep the two configuration so we keep the first: ID0.

About the LoRa message, we notice a **major difference between the estimation of the airtime and the reality** of this metric. The special measurement 1.1 has also demonstrated that this error varies according to the spreading factor parameter. We definitely must take into account this information for the total estimation.

As expected, **the static current varies depending on the motherboard**. The special measurement 1.2 allows us to obtain an average value of the static current and a value for the error (= ±1.26 µA).

<u>Date of measurements :</u> 20/01/2021

2. The buttons board

2.1 Description

The buttons board is a sensor fitted with four buttons. When the board is connected to the motherboard, a press on one of the buttons provokes an interrupt towards the motherboard. According to the button pushed, the motherboard can send a message to the TTN console. This board is based on the microcontroller PIC16F18446.

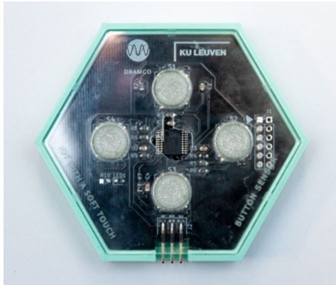

Figure 7 The buttons board

This board requires no specific configurations via the IWAST configurator. We can't set a polling interval and there are no thresholds.

2.2 Hardware analysis

<u>PIC16F18446 - Main MCU</u>

This sensor board is built around the 16-bit PIC16F18446. Its wiring is classic and respects all specifications from the Datasheet. The power supply is set to 3.3 V and all decoupling capacitors respect the recommended values of the Datasheet.

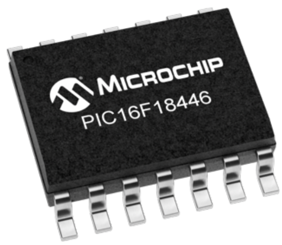

Figure 8 PIC16F18446

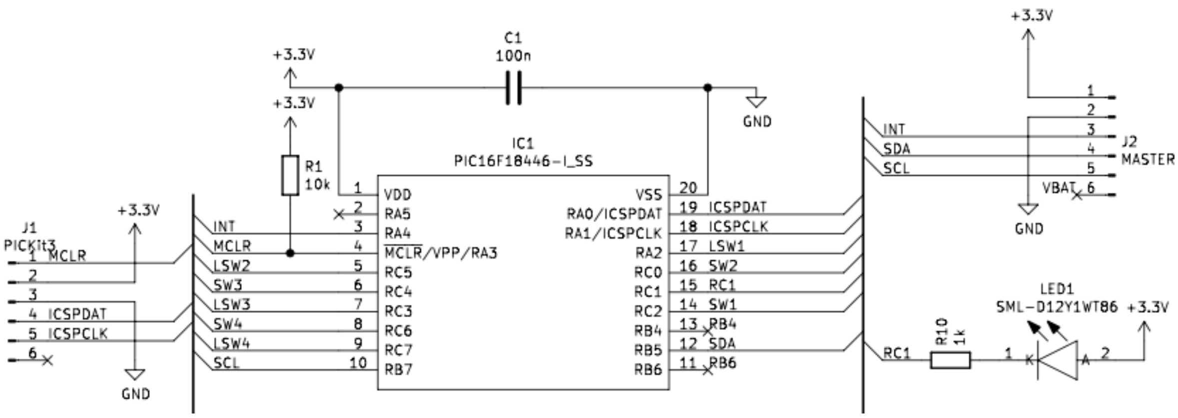

Figure 9 Wiring of the PIC16F18446

We can estimate the current consumption of the MCU depending on the power mode. In the code, we quote that the bit VREGCON in the register VREGPM is set to 1. So, the Low-Power sleep mode is enabled when the MCU pass to the sleep mode. We can determine the total current consumption according to the Datasheet[8] :

PIC16F18426/46 only									
Standard Operating Conditions (unless otherwise stated), VREGPM = 1									
Param. No.	Sym.	Device Characteristics	Min.	Typ.†	Max. +85°C	Max. +125°C	Units	Conditions	
								V_{DD}	Note
D200	I_{PD}	I_{PD} Base	—	0.40	2.5	8	μA	3.0V	
D200A			—	18	25	30	μA	3.0V	VREGPM = 0
D201	I_{PD_WDT}	Low-Frequency Internal Oscillator/WDT	—	1.0	2.9	9	μA	3.0V	
D202	I_{PD_SOSC}	Secondary Oscillator (S_{OSC})	—	1.2	4.3	9.2	μA	3.0V	
D203	I_{PD_FVR}	FVR	—	40	69	70	μA	3.0V	
D204	I_{PD_BOR}	Brown-out Reset (BOR)	—	11	16	19	μA	3.0V	
D207	I_{PD_ADCA}	ADC - Non-converting	—	0.38	2.5	8.0	μA	3.0V	ADC not converting (4)
D208	I_{PD_CMP}	Comparator	—	31	58	59	μA	3.0V	

Table 4 Typical Current Consumption in Power-Down mode

The current consumption for the active mode is more difficult to estimate due to the particular clock using. We need to analyse the code to evaluate which peripherals are enabled, when and how much time they are active. Outside of the sleep mode, the Datasheet is not very clear about the current consumption of each peripherals. We definitely need to analyse this mode with real measurements.

LED

The LEDs used for the four buttons are part of the short travel key switch from MARQUARDT®. We use yellow LEDs and the Datasheet[9] gives us the forward voltage: 1.8 V. So, with serial resistors R3, R5, R7 and R9 (each 1 kΩ) and the power supply (3.3 V), it's easy to determine the current that will pass through when the LED is active:

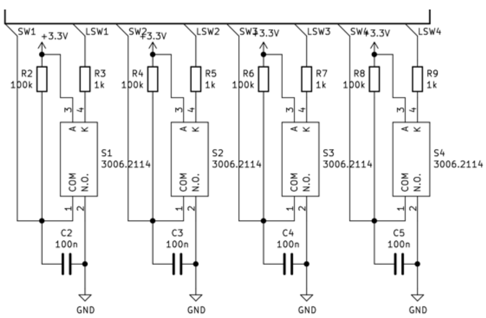

Figure 10 Wiring of the four buttons

$$I\,[A] = \frac{U\,[V]}{R\,[\Omega]} = \frac{3.3 - 1.8}{1000} = 1.5\,mA$$

[8] Table from the Datasheet "*PIC16(L)F18426/46 - 14/20-Pin Full-Featured, Low Pin Count Microcontrollers with XLP*" – p. 636
[9] From the Datasheet "MARQUARDT® Technical Specification K30062000" - p.17

2.3 Measurements

Modus operandi

1. Plug the motherboard, the buttons board and the *Otii ARC* according to the scheme fig. 11 (refer to annex A in order to understand how to configure the measuring device *Otii ARC*).

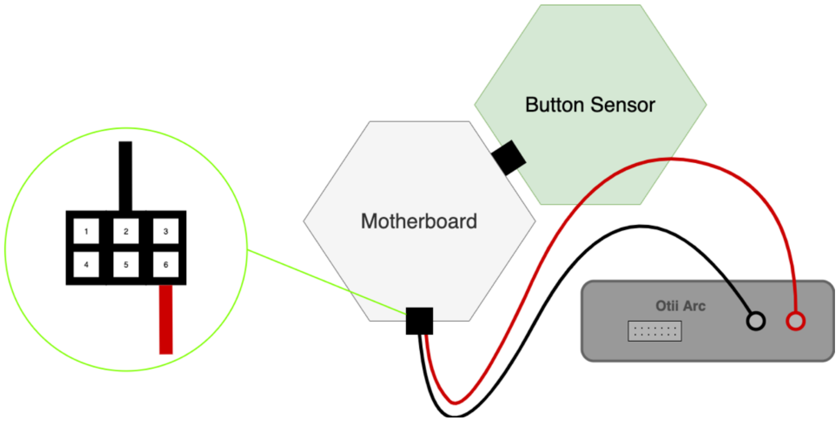

Figure 11 Connect the motherboard, the buttons board and the Otii Arc (LOC 1)

2. Set the system configurations via the *IWAST configurator app* (refer to annex B in order to understand how to use this application). One configuration must be set:

   2.1 **Buttons without accumulation data (ID 2)**:
   - The accumulation data feature must be disabled
   - The polling interrupts must be disabled
   - The thresholds (high and low) must be disabled

3. Collect the current measurements for each configuration at different moment. The test should last 15 min to be relevant. With this physical configuration, we can collect the values of the current flowing between the power supply and the motherboard (location 1) and between the power supply and the buttons module (location 2) (Figure 12). Multiple scenarios must be analysed with this module:

   A. When nothing happens (static power / sleep mode)
   B. When there is a push on a button (dynamic power 2)
   C. When data are sent by the motherboard (data sending power -> Normal Message)

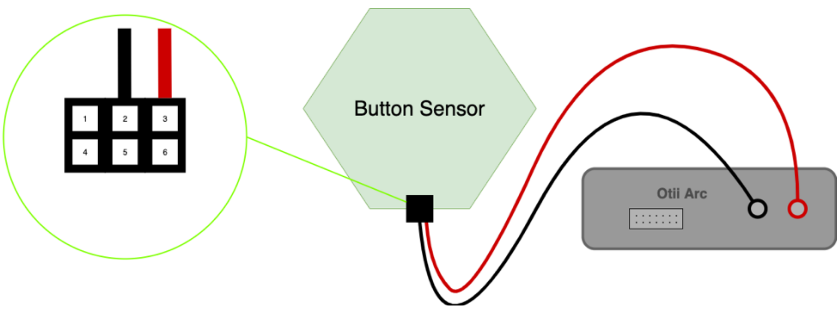

Figure 12 Connect the button sensor and the Otii Arc (LOC 2)

4. Collect measurements for special cases. Multiple scenarios must be analysed:

Special Meas. 2.1 : ***Accumulated message save (for one metric)***

This special measurement must be taken to determine the power consumption characteristic of a saving operation for data accumulation. We analyse the power consumption for the saving of one metric (after a push on a button).

Special Meas. 2.2 : ***Data sent by the motherboard (accumulated message)***

This special measurement must be taken to determine the power consumption when data are sent by the motherboard if the data accumulation is enable. We need to analyse the same parameters than a normal message.

Special Meas. 2.3: ***DATA_TX Time variation depending on the spreading factor (Normal Message) -> Payload length = 4 bytes***

This special measurement must be taken to determine if the LoRa airtime estimation (determined by an estimation online tool[10]) is correct or not. So we collect the airtime of the LoRa castle for each possible spreading factor (7 to 12) and we note the difference between the estimation and the reality. This test must be performed with a normal message consisting of a 4-byte payload.

Special Meas. 2.4: ***DATA_TX Time variation depending on the spreading factor (Accumulated Message) -> Payload length = 30 bytes***

This special measurement must be taken to determine if the LoRa airtime estimation (determined by an estimation online tool) is correct or not. So we collect the airtime of the LoRa castle for each possible spreading factor (7 to 12) and we note the difference between the estimation and the reality. This test must be performed with an accumulated message consisting of a 30-byte payload.

[10] Link for the tool: https://www.loratools.nl/#/airtime

Results

Configuration 2.1 : Buttons without accumulation data (ID 2)

Scenario A - Nothing happens (static power / sleep mode)

Location 2	Average
Maximum Value	1.104 µA
Minimum Value	-0.4932 µA
Average Value	0.2982 µA

Measure	1	2	3	4	5
Max. Value	1.15	1.11	1.14	1.08	1.04
Min. Value	-0.524	-0.524	-0.572	-0.397	-0.449
Aver. Value	0.321	0.293	0.283	0.308	0.286

Remarks :

This value is correct but we can't take it for the estimations. As a matter of fact, we measure a static current of 26,6 µA for the motherboard alone and 27,4 µA with the two boards connected. So we are supposed to have a static current of 0,8 µA for the sensor board alone. We must use this value (0,8) if we want a correct estimation of the total consumption.

Scenario B - Push on a button (dynamic power 2)

Location 1	Average
Maximum Value	12740 µA
Minimum Value	-292.8 µA
Average Value	6658 µA
Energy	10.286 µWh
Interval Time	1687.54 ms

Measure	1	2	3	4	5
Max. Value	14000	13100	13100	11300	12200
Min. Value	-294	-295	-295	-291	-289
Aver. Value	6930	6860	6870	6220	6410
Energy	10.7	10.6	10.6	9.62	9.91
Interval Time	1688.8	1688.5	1686.5	1687	1686.9

Remarks :

Scenario C - Data sent by the motherboard (data sending power -> Normal Message)

Location 1	Average
WUT_ST Time	77.3 ms
WUT_ST Energy	1.05 µWh
DATA_TX Time	662.82 ms
DATA_TX Aver. Cur	47080 µA
DATA_RX Time	2257.3 ms
DATA_RX Energy	37.36 µWh

Remarks :

We are supposed to have DATA_TX Time = 413.7 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 249.12 ms

Measure	1	2	3	4	5
WUT_ST Time	77.2	78.3	76.9	77.3	76.8
WUT_ST Energy	1.05	1.06	1.05	1.05	1.04
DATA_TX Time	662	663.2	663	662.8	663.1
DATA_TX Aver. Cur	47100	47100	47100	47100	47000
DATA_RX Time	2257.9	2255.7	2259.4	2256.6	2257.12
DATA_RX Energy	37.4	37.3	37.4	37.3	37.4

Special Measurements

Special Measurement 2.1 - Accumulated message save (for one metric)

Location 1	Average
Maximum Value	16320 µA
Minimum Value	-228.6 µA
Average Value	9208 µA
Energy	0.3714 µWh
Interval Time	44.02 ms

Measure	1	2	3	4	5
Max. Value	16300	16300	16400	16300	16300
Min. Value	-161	-231	-283	-168	-300
Aver. Value	9200	9250	9230	9200	9160
Energy	0.371	0.374	0.369	0.371	0.372
Interval Time	44	44.1	43.7	44	44.3

Remarks :

There are strange effect before the first event WUM. The time to save data is multiplied by 2 so the energy too. It only happens a few moment so we can neglect this side effect.

Special Measurement 2.2 - Data sent by the motherboard (accumulated message)

Location 1	Average
WUT Time	134.24 ms
WUT Energy	1.876 µWh
DATA_TX Time	1237.46 ms
DATA_TX Aver. Cur	47060 µA
DATA_RX Time	2255.2 ms
DATA_RX Energy	38.7 µWh

Remarks :

We are supposed to have DATA_TX Time = 905.22 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 332.24 ms

Measure	1	2	3	4	5
WUT_ST Time	134.5	134.3	134.4	133.2	134.8
WUT_ST Energy	1.88	1.88	1.88	1.86	1.88
DATA_TX Time	1237	1237.1	1237.2	1237.9	1238.1
DATA_TX Aver. Cur	47500	47700	47400	46700	46000
DATA_RX Time	2251.7	2257.2	2257.2	2252.77	2257.1
DATA_RX Energy	38.8	38.8	38.7	38.7	38.5

Special Measurement 2.3 - DATA_TX Time variation depending on the spreading factor (Normal Message) -> Payload length = 4 bytes			
Location 1	Measure	Theory	Delta
DATA_TX Time (SF = 7)	55.1 ms	30.98 ms	24.12 ms
DATA_TX Time (SF = 8)	97 ms	61.95 ms	35.05 ms
DATA_TX Time (SF = 9)	169.4 ms	123.9 ms	45.5 ms
DATA_TX Time (SF = 10)	333 ms	206.85 ms	126.15 ms
DATA_TX Time (SF = 11)	662.82 ms	413.7 ms	249.12 ms
DATA_TX Time (SF =12)	1323 ms	827.39 ms	495.61 ms
Remarks :			

Special Measurement 2.4 - DATA_TX Time variation depending on the spreading factor (Accumulated Message) -> Payload length = 30 bytes			
Location 1	Measure	Theory	Delta
DATA_TX Time (SF = 7)	102 ms	71.94 ms	30.06 ms
DATA_TX Time (SF = 8)	179.1 ms	123.39 ms	55.71 ms
DATA_TX Time (SF = 9)	333.3 ms	226.3 ms	107 ms
DATA_TX Time (SF = 10)	579.9 ms	452.61 ms	127.29 ms
DATA_TX Time (SF = 11)	1237.46 ms	905.22 ms	332.24 ms
DATA_TX Time (SF =12)	2307.9 ms	1646.59 ms	661.31 ms
Remarks :			

2.2 Conclusion

The remark from the measurement of the scenario A (collect the static current) has shown that we can't just measure the static current of the buttons board alone.

About the LoRa message, we notice again that the LoRa airtime estimation is not correct. For the special measurement 2.3, we have approximately the same result than the special measurement 1.1. It's probably due to the payload lengths which are quite similar (difference of 3 byte). However, the special measurement 2.4 with a payload length of 30 byte has shown that the delta error changes significantly.

To conclude these series of tests on LoRa messages (1.1, 2.3 and 2.4), we must take into account the delta error of the tests 1.1 and 2.3 for a normal message estimation and the delta error of the test 2.4 for an accumulated message estimation. In other words, when we need to calculate the LoRa airtime for the total estimation, we can calculate the airtime with the estimation tool (from the website) and just add the delta error to obtain a value close to what can be observed on IWAST.

<u>Date of measurements :</u> 20/01/2021

3. The power board

2.1 Description

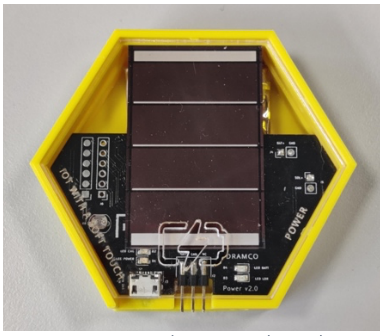

Figure 13 The Power board

The power board allows the IWAST system to be self-sufficient thanks to a battery and a solar panel combined on a single board. This board has the same size as a common sensor board and permanently delivers 3,3 V directly to the motherboard via the VBAT pin. This board is based on the microcontroller PIC16F18446.

To obtain a correct voltage output value, the chip BQ25570 is used to extract all the power from the solar panel and to regulate this one to a constant voltage of 3,3 V. Nevertheless, if the battery is completely discharged, there is still a possibility to charge it by a micro USB port. This solution is possible thanks to the chip BQ24040 that convert a 5V supply to a proper battery voltage.

Multiple configurations can be set via the IWAST configurator to obtain data from the power board. We can have the voltage of the battery or get a measure of the light that coming on the board thanks to a measure of a Light Dependent Resistor (LDR) voltage (included on the board). These data can be acquired thanks to a polling interruption (sent by the motherboard) or by setting threshold levels (high and low).

For the battery voltage, we must multiply the value of the voltage thresholds by a factor 10 000 in the IWAST configurator. For instance, if we want to set the high threshold level to 3,7 V, we have to encode 37000. For the LDR voltage, multiple "lux" values depending on luminosity situations are described in the IWAST documentation[11].

2.2 Hardware analysis

PIC16F18446 - Main MCU

This sensor board is built around the 16-bit PIC16F18446. Its wiring is classic and respects all specifications from the Datasheet. The power supply is set to 3.3 V and all decoupling capacitors respect the recommended values of the Datasheet.

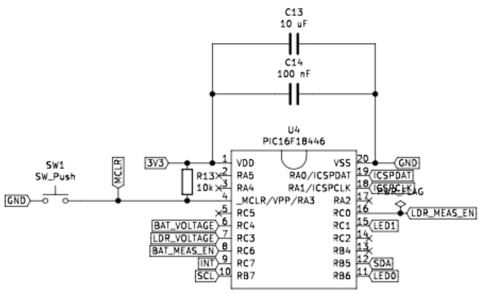

Figure 14 Wiring of the PIC16F18446

[11] Address : https://dramco-iwast.github.io/docs/sensor-boards/powermodule.html

We can estimate the current consumption of the MCU depending on the power mode. In the code, we quote that the bit VREGCON in the register VREGPM is set to 1. So, the Low-Power sleep mode is enabled when the MCU pass to the sleep mode. We can determine the total current consumption according to the Datasheet[12] :

PIC16F18426/46 only									
Standard Operating Conditions (unless otherwise stated), VREGPM = 1									
Param. No.	Sym.	Device Characteristics	Min.	Typ.†	Max. +85°C	Max. +125°C	Units	Conditions	
								V_{DD}	Note
D200	I_{PD}	I_{PD} Base	—	0.40	2.5	8	μA	3.0V	
D200A			—	18	25	30	μA	3.0V	VREGPM = 0
D201	I_{PD_WDT}	Low-Frequency Internal Oscillator/WDT	—	1.0	2.9	9	μA	3.0V	
D202	I_{PD_SOSC}	Secondary Oscillator (S_{OSC})	—	1.2	4.3	9.2	μA	3.0V	
D203	I_{PD_FVR}	FVR	—	40	69	70	μA	3.0V	
D204	I_{PD_BOR}	Brown-out Reset (BOR)	—	11	16	19	μA	3.0V	
D207	I_{PD_ADCA}	ADC - Non-converting	—	0.38	2.5	8.0	μA	3.0V	ADC not converting (4)
D208	I_{PD_CMP}	Comparator	—	31	58	59	μA	3.0V	

Table 5 Typical Current Consumption in Power-Down mode

The current consumption for the active mode is more difficult to estimate due to the particular clock using. We need to analyse the code to evaluate which peripherals are enabled, when and how much time they are active. Outside of the sleep mode, the Datasheet is not very clear about the current consumption of each peripherals. We definitely need to analyse this mode with real measurements.

BQ25570 - Boost charger and buck converter

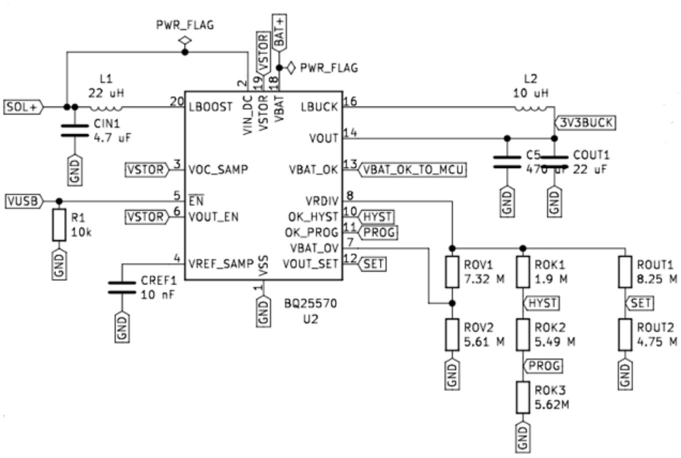

Figure 15 Wiring of the BQ25570

This component is used by the Power module to collect the energy from the solar panel, to charge the battery and to generate a regulated voltage source of 3.3 V from the battery output. It is possible to set the output voltage thanks to the two resistors R_{OUT1} and R_{OUT2}.

The other resistors are used to set the other parameters: the battery overvoltage protection (R_{OV1} et R_{OV2}) and the operating range of the battery voltage (R_{OK1}, R_{OK2} et R_{OK3}).

[12] Table from the Datasheet "*PIC16(L)F18426/46 - 14/20-Pin Full-Featured, Low Pin Count Microcontrollers with XLP*" – p. 636

It consists of a nano power boost charger and buck converter in charge of supplying all the energy for the Power board (and the complete IWAST system). According to the datasheet[13], this chip has a **typical quiescent current of 488 nA during its full operating mode**.

BQ24040 - USB charger

This unit is used if we want to charge the battery directly from a USB connection. It is specifically designed to properly charge Li-Ion and Li-Pol Battery. The two associated LEDs are used to notify the users if the battery is charging or has finishing charging. This chip is supposed to be useless during normal operation of the IWAST system (the battery must be charged only by the solar panel).

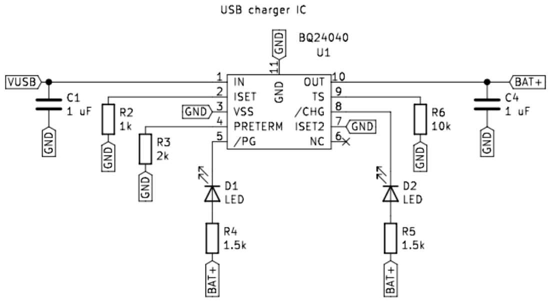

Figure 16 Wiring of the BQ24040

According to the Datasheet[14], the typical active supply current for the chip is equal to 0.8 mA.

TPS22860 - Switch

The power board possess three switches in order to disabled some part of the board. The first is used to enable/disable the power supply that come from the BQ25570 chip. If the solar panel can provide enough energies to allow the BQ25570 to produce 3.3 V, the switch is enabled. The second switch is used to enable/disable the possibility to measure the voltage of the battery. The last provides the ability to enable/disable the measurement of the LDR voltage (LDR = Light Dependant Resistor).

The special feature of these switches is that they are “ultra-low leakage”. It seems that they consume very little power when they are ON. We can notice this behaviour in the Datasheet[15]:

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
POWER SUPPLIES AND CURRENTS							
$I_{Q,\ VBIAS}$	V_{BIAS} quiescent current	$I_{OUT} = 0$, $V_{IN} = V_{ON} = V_{BIAS} = 3.3$ V			10	100	nA
$I_{SD,\ VBIAS}$	V_{BIAS} shutdown current	$V_{ON} = 0$ V			10	100	nA
$I_{SD,\ VIN}$	V_{IN} shutdown current	$V_{ON} = 0$ V, $V_{OUT} = 1$ V	$V_{IN} = 3.0$ V		2	50	nA
I_{ON}	ON pin input leakage current	$V_{ON} = 5.5$ V				100	nA

Table 6 Electrical characteristics of the switch TPS22860

TPS705 - Low-Dropout Regulator

This Low-Dropout Regulator was placed on the board to regulate the voltage output of the battery if the BQ25570 doesn’t work. It was used for the development part of the board and shouldn’t be present for the final version of the system. So, we don’t need to take it into account for the estimation of the power consumption.

---

[13] From the Datasheet “*bq25570 nano power boost charger and buck converter for energy harvester powered applications*” - p. 6
[14] From the Datasheet “*BQ2404x 1A, Single-Input, Single Cell Li-Ion and Li-Pol Battery Charger With Auto Start*” - p. 7
[15] Table from the Datasheet “*TPS22860 Ultra-Low Leakage Load Switch*” - p. 4

2.3 Measurements

Modus operandi

1. Plug the motherboard, the power board and the *Otii ARC* according to the scheme fig. 17 (refer to annex A in order to understand how to configure the measuring device *Otii ARC*). With this sensor board, we must remove the solar panel and the battery (the battery is replaced by the *Otii ARC*).

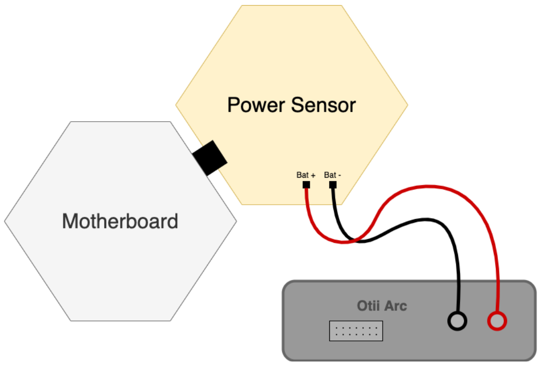

Figure 17 Connect the motherboard, the power sensor and the Otii Arc

2. Set the system configurations via the *IWAST configurator app* (refer to annex B in order to understand how to use this application). Several configurations must be set:

2.1 **Power board without accumulation data, without polling and without thresholds (ID 3)**:
- The accumulation data feature must be disabled
- The polling interrupts must be disabled
- The thresholds (high and low) must be disabled

2.2 **Power board without accumulation data, without polling but with thresholds (ID 4)**:
- The accumulation data feature must be disabled
- The polling interrupts must be disabled
- The thresholds (high and low) must be enabled
- The threshold high for the metric 1 must be equal to 36000
- The threshold low for the metric 1 must be equal to 31000
- The threshold high for the metric 2 must be equal to 2000
- The threshold low for the metric 2 must be equal to 0
- We must set the power supply from the *Otii ARC* at 3.3 V
- The system must be placed in a normally lighted room (not in the sunshine)

2.3 **Power board without accumulation data, without thresholds but with polling (ID 5):**
- The accumulation data feature must be disabled
- The polling interrupts must be enabled and set to 2 min
- The thresholds (high and low) must be disabled

2.4 **Power board with accumulation data, polling and thresholds (ID 6):**
- The accumulation data feature must be enabled
- The polling interrupts must be enabled and set to 2 min
- The thresholds (high and low) must be enabled
- The threshold high for the metric 1 must be equal to 36000
- The threshold low for the metric 1 must be equal to 31000
- The threshold high for the metric 2 must be equal to 2000
- The threshold low for the metric 2 must be equal to 0
- We must set the power supply from the *Otii ARC* at 3.3 V
- The system must be placed in a normally lighted room (not in the sunshine)

3. Collect the current measurements for each configuration at different moment. The test should last 15 min to be relevant. With this physical configuration, we can collect the values of the current flowing between the power supply provided by the *Otii ARC* and the power board (location 1). Multiple scenarios must be analysed with this module (depending on the configuration, multiple scenarios cannot be analysed):

    A. When nothing happens (static power / sleep mode)
    B. When a threshold event (not exceeded) occurs (dynamic power 1)
    C. When a threshold event (exceeded) occurs (dynamic power 2)
    D. When a polling interrupt occurs (dynamic power 3)
    E. When data are sent by the motherboard (data sending power -> Normal Message)

4. Collect measurements for special cases. Multiple scenarios must be analysed:

Special Meas. 3.1 : ***Accumulated message save (for two metrics)***

This special measurement must be taken to determine the power consumption characteristic of a saving operation for data accumulation. We analyse the power consumption for the saving of two metrics (after an event like a threshold exceeded or a polling interrupt).

Special Meas. 3.2 : ***Data sent by the motherboard (accumulated message)***

This special measurement must be taken to determine the power consumption when data are sent by the motherboard if the data accumulation is enable. We need to analyse the same parameters than a normal message.

Special Meas. 3.3: ***Static Power of the Power Module alone***

This special measurement must be taken to evaluate the difference between the results from all scenarios A and the static current from the power board alone.

Results

Configuration 2.1 : Power board without accumulation data, without polling and without thresholds (ID 3)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	320.6 µA
Minimum Value	-13.6 µA
Average Value	29.86 µA

Measure	1	2	3	4	5
Max. Value	349	346	281	289	338
Min. Value	-9.27	-24.8	-9.43	-9.61	-15
Aver. Value	29.9	29.9	29.8	29.8	29.9

Remarks :

*So, to obtain the static value of the power board alone, we can do 29,86 µA -27,2 µA and we obtain **2,66 µA** for the static current of the power board alone. (27,2 µA is the static current of the motherboard used for this test)*

Configuration 2.2 : Power board without accumulation data, without polling but with thresholds(ID 4)

Scenario A - Nothing happens (static power / sleep mode)					
Location 1	Average				
Maximum Value	244.6 µA				
Minimum Value	-8.734 µA				
Average Value	30.6 µA				
Measure	1	2	3	4	5
Max. Value	293	288	148	146	348
Min. Value	-10.2	-8.18	-8.41	-8.53	-8.35
Aver. Value	30.5	30.6	30.6	30.7	30.6

Remarks :

*So, to obtain the static value of the power board alone, we can do 30,6 µA - 27,2 µA and we obtain **3,4 µA** for the static current of the power board alone. (27,2 µA is the static current of the motherboard used for this test)*

Scenario B - Threshold event (not exceeded) (dynamic power 1)					
Location 1	Average				
Maximum Value	9802 µA				
Minimum Value	-678.4 µA				
Average Value	5988 µA				
Energy	0.179 µWh				
Interval Time	32.46 ms				
Measure	1	2	3	4	5
Max. Value	9250	8920	12800	10100	7940
Min. Value	-895	-603	-627	-701	-566
Aver. Value	6050	5920	6240	5850	5880
Energy	0.18	0.178	0.18	0.179	0.178
Interval Time	32.3	32.6	31.3	33.2	32.9

Remarks :

Scenario C - Threshold event (exceeded) (dynamic power 2)

Location 1	Average
Maximum Value	9250 µA
Minimum Value	-18.84 µA
Average Value	7154 µA
Energy	0.184 µWh
Interval Time	27.9 ms

Measure	1	2	3	4	5
Max. Value	10600	8350	7610	11100	8590
Min. Value	-35	-35	45.8	-35	-35
Aver. Value	7140	7140	7120	7220	7150
Energy	0.186	0.185	0.184	0.18	0.185
Interval Time	28.2	28	28	27.1	28.2

Remarks :

Scenario E - Data sent by the motherboard (data sending power -> Normal Message)

Location 1	Average
WUT Time	77.104 ms
WUT Energy	1.066 µWh
DATA_TX Time	745.38 ms
DATA_TX Aver. Cur	46860 µA
DATA_RX Time	2257.98 ms
DATA_RX Energy	38.66 µWh

Remarks :

We are supposed to have DATA_TX Time = 495.62 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 249.76 ms

Measure	1	2	3	4	5
WUT Time	77.38	77.25	76.84	76.18	77.87
WUT Energy	1.07	1.07	1.06	1.06	1.07
DATA_TX Time	745.3	745.2	745.6	745	745.8
DATA_TX Aver. Cur	46000	47000	47200	47100	47000
DATA_RX Time	2260	2259	2258.7	2251.71	2260.5
DATA_RX Energy	38.6	38.6	38.7	38.7	38.7

Configuration 2.3 : Power board without accumulation data, without thresholds but with polling (ID 5)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	386.8 µA
Minimum Value	-8.794 µA
Average Value	30.48 µA

Measure	1	2	3	4	5
Max. Value	385	386	385	392	386
Min. Value	-8.16	-8.2	-8.42	-8.79	-10.4
Aver. Value	30.6	30.5	30.5	30.4	30.4

Remarks :

*So, to obtain the static value of the power board alone, we can do 30,48 µA -27,89 µA and we obtain **2,59 µA** for the static current of the power board alone. (27,89 µA is the static current of the motherboard used for this test)*

Scenario D - Polling interrupt (dynamic power 3)

Location 1	Average
Maximum Value	9698 µA
Minimum Value	42.36 µA
Average Value	6640 µA
Energy	0.044 µWh
Interval Time	7.18 ms

Measure	1	2	3	4	5
Max. Value	9800	9840	9210	9860	9780
Min. Value	42.1	46.2	43.5	45.5	34.5
Aver. Value	6840	6500	6240	6880	6740
Energy	0.045	0.04	0.044	0.045	0.046
Interval Time	7.1	6.8	7.6	7	7.4

Remarks :

Polling without data accumulation doesn't work well…

Scenario E - Data sent by the motherboard (data sending power -> Normal Message)

Location 1	Average
WUT_ST Time	77.06 ms
WUT_ST Energy	1.084 µWh
DATA_TX Time	744.62 ms
DATA_TX Aver. Cur	47500 µA
DATA_RX Time	2257.59 ms
DATA_RX Energy	39 µWh

<u>Remarks :</u>

We are supposed to have DATA_TX Time = 495.62 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 249 ms

Measure	1	2	3	4	5
WUT_ST Time	77.06	77.06	77.09	77	77.09
WUT_ST Energy	1.09	1.07	1.08	1.09	1.09
DATA_TX Time	744	745.5	745.3	743.2	745.1
DATA_TX Aver. Cur	47700	47800	47300	47100	47600
DATA_RX Time	2253.06	2257.7	2259	2260	2258.2
DATA_RX Energy	39	39	39	39	39

Configuration 2.4 : Power board with accumulation data, polling and thresholds (ID 6)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	627 µA
Minimum Value	-9.89 µA
Average Value	31.36 µA

Measure	1	2	3	4	5
Max. Value	999	403	1000	400	333
Min. Value	-10.4	-8.48	-12.1	-10	-8.45
Aver. Value	31.5	31.4	31.4	31.3	31.2

Remarks :
*So, to obtain the static value of the power board alone, we can do 31,36 µA -27,89 µA and we obtain **3,47 µA** for the static current of the power board alone. (27,89 µA is the static current of the motherboard used for this test)*

Scenario B - Threshold event (not exceeded) (dynamic power 1)

Location 1	Average
Maximum Value	11250 µA
Minimum Value	-733 µA
Average Value	6385 µA
Energy	0.181 µWh
Interval Time	30.85 ms

Measure	1	2	3	4	5
Max. Value	10200	12300	/	/	/
Min. Value	-731	-735	/	/	/
Aver. Value	6370	6400	/	/	/
Energy	0.181	0.181	/	/	/
Interval Time	30.9	30.8	/	/	/

Remarks :
It's difficult to collect correct data for this event. In the test, we had only two relevant measurement

Scenario C - Threshold event (exceeded) (dynamic power 2)

Location 1	Average
Maximum Value	9058 µA
Minimum Value	89.62 µA
Average Value	7178 µA
Energy	0.182 µWh
Interval Time	27.54 ms

Measure	1	2	3	4	5
Max. Value	12000	8240	8730	8160	8160
Min. Value	107	93.6	92.5	130	25
Aver. Value	7310	7040	7160	7180	7200
Energy	0.183	0.182	0.179	0.185	0.182
Interval Time	27.1	28.1	27.1	28	27.4

Remarks :

Scenario D - Polling interrupt (dynamic power 3)

Location 1	Average
Maximum Value	9786 µA
Minimum Value	78.22 µA
Average Value	6868 µA
Energy	0.046 µWh
Interval Time	7.21 ms

Measure	1	2	3	4	5
Max. Value	9740	9800	9760	9820	9810
Min. Value	89.9	97.4	31.4	68.4	104
Aver. Value	7010	7060	6670	6790	6810
Energy	0.047	0.047	0.045	0.045	0.046
Interval Time	7.2	7.18	7.22	7.22	7.25

Remarks :

Special Measurements

Special Measurement 3.1 - Accumulated message save (for two metrics)

Location 1	Average
Maximum Value	18480 µA
Minimum Value	-810.6 µA
Average Value	9284 µA
Energy	0.770 µWh
Interval Time	90.52 ms

Measure	1	2	3	4	5
Max. Value	18500	18500	18400	18500	18500
Min. Value	-683	-924	-787	-768	-891
Aver. Value	9260	9260	9300	9280	9320
Energy	0.768	0.769	0.773	0.769	0.772
Interval Time	90.5	90.7	90.6	90.5	90.3

Remarks :

Special Measurement 3.2 - Data sent by the motherboard (accumulated message)

Location 1	Average
WUT Time	121.6 ms
WUT Energy	1.734 µWh
DATA_TX Time	1155.08 ms
DATA_TX Aver. Cur	47320 µA
DATA_RX Time	2259.32 ms
DATA_RX Energy	39.06 µWh

Remarks :

We are supposed to have DATA_TX Time = 905.22 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 249.86 ms

Measure	1	2	3	4	5
WUT_ST Time	121.7	121.8	121.8	121.9	120.8
WUT_ST Energy	1.74	1.74	1.74	1.74	1.71
DATA_TX Time	1155.4	1155.3	1155.5	1153.9	1155.3
DATA_TX Aver. Cur	48000	47100	47100	47200	47200
DATA_RX Time	2258.1	2260.3	2259.1	2260.5	2258.6
DATA_RX Energy	39.1	39.1	39	39.1	39

Special Measurement 3.3 – Static Power of the Power Module alone

Location 1	Average
Maximum Value	41.52 µA
Minimum Value	-5.16 µA
Average Value	1.25 µA

Measure	1	2	3	4	5
Max. Value	41.9	41.7	41.4	41.5	41.1
Min. Value	-5.19	-5.1	-5.13	-5.19	-5.2
Aver. Value	1.26	1.26	1.26	1.25	1.25

Remarks :

Data not relevant if we consider motherboard + power board

3.1Conclusion

At the end of the tests, **no major differences were noticed between ID3 and ID5**. Therefore, we can conclude that the enabling/disabling of the polling feature has no influence on the static current. However, when we enable/disable the threshold feature, we can notice a major difference for the static current. For the total estimation, we need to create two sections : “Power with no thresholds” (ID3 and ID5) and “Power with thresholds” (ID4 and ID6).

For the total estimation, it’s not valuable to keep these four configurations so **we keep the ID5 (which becomes ID 2 in the excel file) and the ID6 (which becomes ID 3 in the excel file)**.

As expected, data are not relevant if we measure the static current of the power module alone. So we need to measure the static current of the motherboard alone and the static current of the two boards connected to determine the static current of the power module.

At the end of the ID 5 test, we also noticed that the polling doesn’t work very well without the data accumulation feature. So, it’s recommended to always use this option when the power module is used.

Date of measurements : 20/01/2021 - 21/01/2021

4. The sound board

4.1 Description

The sound board is fitted with a microphone to measure the intensity of ambient noise. To limit the power consumption of the sensor, the microphone Vesper VM1010 has been chosen (particularly for its good performance). The microcontroller used for the sensor is a PIC16F18446. We can easily set thresholds levels (high and low) in order to generate interrupts to the motherboard when thresholds are exceeded.

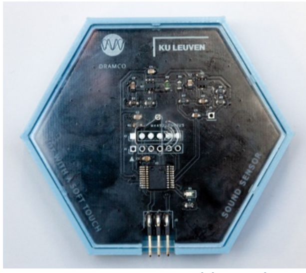

Figure 18 Sound board

When thresholds are enabled, the microphone is ON and sends a message with the noise measure to the microcontroller. If the measure exceeds the threshold level (high or low), an interrupt is generated to the motherboard and the microphone is switched OFF for one minute (to avoid generating a lot of interrupts during a small interval of time). When thresholds are disabled, the microphone is always switched OFF (except if a polling interrupt appears).

We can set a polling interval and two thresholds level (high or low) via the IWAST configurator. However, setting a low threshold is not relevant with this sensor according to the interest of this sensor (so, all measurements describe below is done with a low threshold level equal to 0 dB).

4.2 Hardware analysis

PIC16F18446 - Main MCU

This sensor board is built around the 16-bit PIC16F18446. Its wiring is classic and respects all specifications from the Datasheet. The power supply is set to 3.3 V and all decoupling capacitors respect the recommended values of the Datasheet.

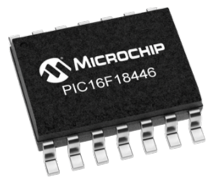

Figure 19 PIC16F18446

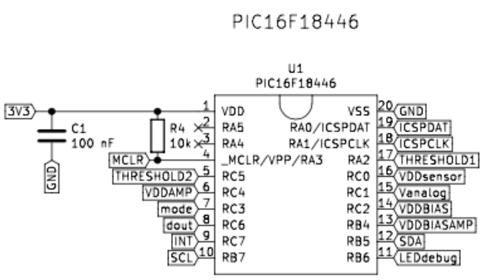

Figure 20 Wiring of the PIC16F18446

We can estimate the current consumption of the MCU depending on the power mode. In the code, we quote that the bit VREGCON in the register VREGPM is set to 1. So, the Low-Power sleep mode is enabled when the MCU pass to the sleep mode. We can determine the total current consumption according to the Datasheet[16] :

PIC16F18426/46 only									
Standard Operating Conditions (unless otherwise stated), VREGPM = 1									
Param. No.	Sym.	Device Characteristics	Min.	Typ.†	Max. +85°C	Max. +125°C	Units	Conditions	
								V_{DD}	Note
D200	I_{PD}	I_{PD} Base	—	0.40	2.5	8	µA	3.0V	
D200A			—	18	25	30	µA	3.0V	VREGPM = 0
D201	I_{PD_WDT}	Low-Frequency Internal Oscillator/WDT	—	1.0	2.9	9	µA	3.0V	
D202	I_{PD_SOSC}	Secondary Oscillator (S_{OSC})	—	1.2	4.3	9.2	µA	3.0V	
D203	I_{PD_FVR}	FVR	—	40	69	70	µA	3.0V	
D204	I_{PD_BOR}	Brown-out Reset (BOR)	—	11	16	19	µA	3.0V	
D207	I_{PD_ADCA}	ADC - Non-converting	—	0.38	2.5	8.0	µA	3.0V	ADC not converting (4)
D208	I_{PD_CMP}	Comparator	—	31	58	59	µA	3.0V	

Table 7 Typical Current Consumption in Power-Down mode

The current consumption for the active mode is more difficult to estimate due to the clock using. We need to analyse the code to evaluate which peripherals are enabled, when and how much time they are active. Outside of the sleep mode, the Datasheet is not very clear about the current consumption of each peripherals. We need to analyse this mode with real measurements.

Amplifier and low-pass filter

The non-inverting amplifier (placed after the high-pass filter) is used to amply the signal with 30 dB. After that, we have a *Sallen-Key* low-pass filter used to reduce the amplitude of the unwanted ultrasonic sound (there is also an amplification of 2,73 dB to amplify the signal to an amplitude which is appropriate for the ADC. All these elements used three TLV341A which are used as operational amplifiers.

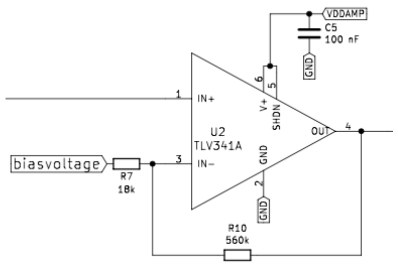

Figure 21 Non-inverting amplifier

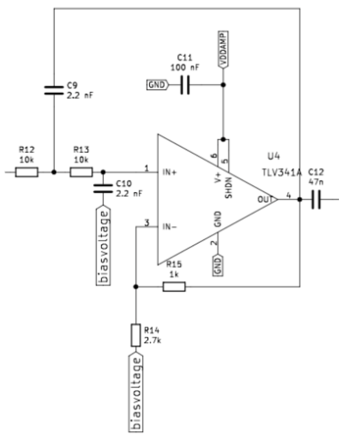

Figure 22 Low-pass filter

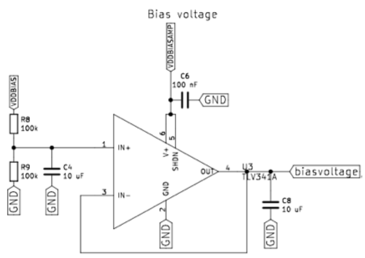

Figure 23 Bias voltage setting

[16] Table from the Datasheet "*PIC16(L)F18426/46 - 14/20-Pin Full-Featured, Low Pin Count Microcontrollers with XLP*" – p. 636

The recommended supply voltage of these op-amps must be between 1,5 V and 5,5 V. According to the schematic of the board, we can determine power supply is set to 2,6 V (voltage from MCU pin). The three op-amps are only active when a measure is done. So, we save a certain amount of current consumption when measurements are disabled. According to the datasheet[17], the current consumption for each op-amp is between 75 and 150 µA. In the shutdown mode, the consumption falls between 10 nA and 1 µA.

Vesper VM1010

This microphone chip is quite particular due to its low-power feature. According to the datasheet[18], it can work in two modes : “Normal” mode (current consumption between 85 and 100 µA) and “Wake on Sound” mode (current consumption between 10 and 14 µA).

The power supply is set to 2,6 V (voltage from MCU pin) and the decoupling capacitor respects the recommended value of the datasheet.

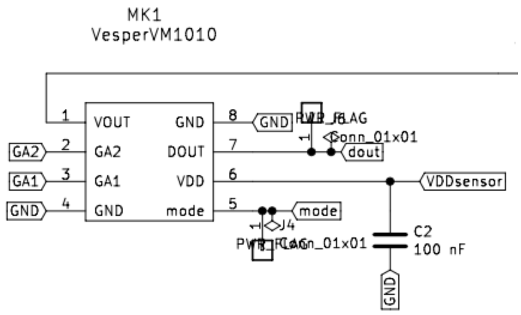

Figure 24 Wiring of the Vesper VM1010

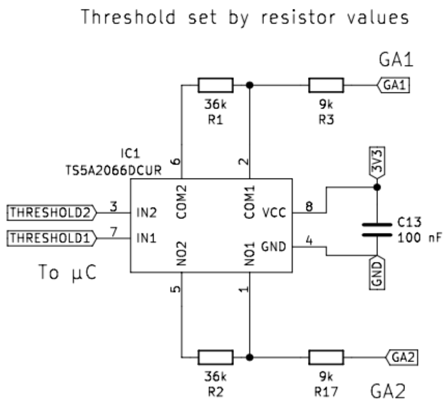

Figure 25 Wiring of the analog switch

We set the threshold levels on pins GA1 and GA2 thanks to multiple values of a resistor. This choice can be achieved with an analog switch, the TSA52066DCUR. According to the datasheet[19], the supply current of this chip is between 0,1 et 1 µA.

The power supply is set to 3,3 V and the decoupling capacitor respects the recommended value of the datasheet.

[17] From the Datasheet “TLV34xxLow-VoltageRail-to-Rail Output CMOSOperational Amplifiers WithShutdown” – p. 7-8
[18] From the Datasheet “VM1010 Low Noise Bottom Port Analog Piezoelectric MEMS Microphone with Wake on Sound” - p. 4
[19] From the Datasheet “TS5A2066 Dual-Channel 10-Ω SPST Analog Switch” - p. 8

LED

The LED used for the debugging feature of the board is blue and connected in serial with a 470 Ω resistor. The typical forward voltage for this kind of LED is equal to 3,2 V and the power supply is set to 3,3 V (voltage from MCU pin). So we can easily determine the maximum current consumption of the LED :

$$I\ [A] = \frac{U\ [V]}{R\ [\Omega]} = \frac{3.3 - 3.2}{470} = 0.21\ mA$$

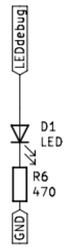

Figure 26 Wiring of the debug LED

4.3 Measurements

Modus operandi

1. Plug the motherboard, the sound board and the *Otii ARC* according to the scheme fig. 27 (refer to annex A in order to understand how to configure the measuring device *Otii ARC*).

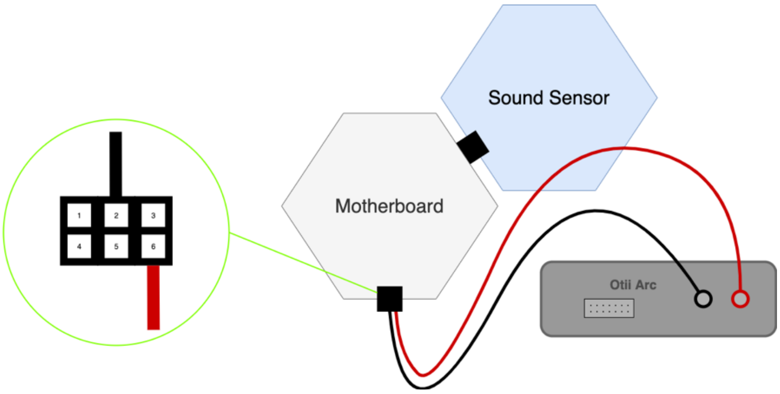

Figure 27 Connect the motherboard, the sound board and the Otii Arc (LOC 1)

2. Set the system configurations via the *IWAST configurator app* (refer to annex B in order to understand how to use this application). Several configurations must be set:

   2.1 **Sound board without accumulation data, without polling and without thresholds (ID 7)**:
   - The accumulation data feature must be disabled
   - The polling interrupts must be disabled
   - The thresholds (high and low) must be disabled

   2.2 **Sound board without accumulation data, without polling but with thresholds (ID 8)**:
   - The accumulation data feature must be disabled
   - The polling interrupts must be disabled
   - The thresholds (high and low) must be enabled
   - The threshold high for the metric 1 must be equal to 80
   - The threshold low for the metric 1 must be equal to 0

   2.3 **Sound board without accumulation data, without thresholds but with polling (ID 9):**
   - The accumulation data feature must be disabled
   - The polling interrupts must be enabled and set to 2 min
   - The thresholds (high and low) must be disabled

   2.4 **Sound board with accumulation data, polling and thresholds (ID 10):**
   - The accumulation data feature must be enabled
   - The polling interrupts must be enabled and set to 2 min
   - The thresholds (high and low) must be enabled
   - The threshold high for the metric 1 must be equal to 80
   - The threshold low for the metric 1 must be equal to 0

3. Collect the current measurements for each configuration at different moment. The test should last 15 min to be relevant. With this physical configuration, we can collect the values of the current flowing between the power supply and the motherboard (location 1). Multiple scenarios must be analysed with this module:

   A. When nothing happens (static power / sleep mode)
   B. When a threshold event (not exceeded) occurs (dynamic power 1)

C. When a threshold event (exceeded) occurs (dynamic power 2)
D. When a polling interrupt occurs (dynamic power 3)
E. When data are sent by the motherboard (data sending power -> Normal Message)

4. Collect measurements for special cases. Multiple scenarios must be analysed:

Special Meas. 4.1 : ***Accumulated message save (for one metric)***

This special measurement must be taken to determine the power consumption characteristic of a saving operation for data accumulation. We analyse the power consumption for the saving of one metric (after an event like a threshold exceeded or a polling interrupt).

Special Meas. 4.2 : ***Data sent by the motherboard (accumulated message)***

This special measurement must be taken to determine the power consumption when data are sent by the motherboard if the data accumulation is enable. We need to analyse the same parameters than a normal message.

Results

Configuration 2.1 : Sound board without accumulation data, without polling and without thresholds (ID 7)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	375.8 µA
Minimum Value	27.38 µA
Average Value	28.68 µA

Measure	1	2	3	4	5
Max. Value	401	400	402	338	338
Min. Value	27.6	27.4	27.4	27.2	27.3
Aver. Value	28.9	28.7	28.7	28.6	28.5

Remarks :

*So, to obtain the static value of the sound board alone, we can do 28,68 µA -27,89 µA and we obtain **0,79 µA** for the static current of the sound board alone. (27,89 µA is the static current of the motherboard used for this test)*

Configuration 2.2 : Sound board without accumulation data, without polling but with thresholds(ID 8)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	294.4 µA
Minimum Value	49.3 µA
Average Value	50.5 µA

Measure	1	2	3	4	5
Max. Value	518	235	239	240	240
Min. Value	49.4	49.1	49.3	49.4	49.3
Aver. Value	50.7	50.4	50.4	50.5	50.5

Remarks :

CAUTION ! *When an exceeded threshold event occurs, the microphone is switched OFF. The average current consumption switches from 50.5 µA to 28.9 µA during 64000 ms.*

So, to obtain the static value of the sound board alone, we can do 50,5 µA -27,89 µA and we obtain ***22,61 µA*** *for the static current of the sound board alone. (27,89 µA is the static current of the motherboard used for this test)*

Scenario B - Threshold event (not exceeded) (dynamic power 1)

Location 1	Average
Maximum Value	7032 µA
Minimum Value	-47.1 µA
Average Value	4068 µA
Energy	1.734 µWh
Interval Time	464.98 ms

Measure	1	2	3	4	5
Max. Value	6980	7040	7080	7070	6990
Min. Value	-41.5	-61.1	-26.3	-49.1	-57.5
Aver. Value	4060	4060	4070	4080	4070
Energy	1.74	1.73	1.74	1.73	1.73
Interval Time	466.4	465	466.3	462.5	464.7

Remarks

Scenario C - Threshold event (exceeded) (dynamic power 2)

Location 1	Average
Maximum Value	7026 µA
Minimum Value	52.54 µA
Average Value	4130 µA
Energy	1.684 µWh
Interval Time	445.48 ms

Measure	1	2	3	4	5
Max. Value	6990	7020	6980	7090	7050
Min. Value	84.6	58.1	50.3	19.6	50.1
Aver. Value	4140	4140	4130	4120	4120
Energy	1.69	1.69	1.68	1.68	1.68
Interval Time	444.9	446.6	445.4	444.7	445.8

Remarks

Scenario E - Data sent by the motherboard (data sending power -> Normal Message)

Location 1	Average
WUT Time	77.06 ms
WUT Energy	1.066 µWh
DATA_TX Time	663.46 ms
DATA_TX Aver. Cur	47880 µA
DATA_RX Time	2257.48 ms
DATA_RX Energy	39.12 µWh

Remarks :

We are supposed to have DATA_TX Time = 413.7 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 249.76 ms

Measure	1	2	3	4	5
WUT Time	76.5	77.1	79.2	76.2	76.3
WUT Energy	1.06	1.06	1.1	1.05	1.06
DATA_TX Time	664.6	663.1	663.3	663.4	662.9
DATA_TX Aver. Cur	48000	48000	47900	47700	47800
DATA_RX Time	2257.2	2258.3	2255.9	2257.3	2258.7
DATA_RX Energy	39.1	39.1	39.1	39.2	39.1

Configuration 2.3 : Sound board without accumulation data, without thresholds but with polling (ID 9)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	411.8 µA
Minimum Value	27.36 µA
Average Value	28.64 µA

Measure	1	2	3	4	5
Max. Value	401	413	413	414	418
Min. Value	27.5	27.4	27.3	27.3	27.3
Aver. Value	28.7	28.7	28.6	28.6	28.6

Remarks :

*So, to obtain the static value of the sound board alone, we can do 28,64 µA -27,89 µA and we obtain **0,75 µA** for the static current of the sound board alone. (27,89 µA is the static current of the motherboard used for this test)*

Scenario D - Polling interrupt (dynamic power 3)

Location 1	Average
Maximum Value	15500 µA
Minimum Value	59.7 µA
Average Value	5352 µA
Energy	2.3 µWh
Interval Time	468.83 ms

Measure	1	2	3	4	5
Max. Value	15500	15500	15500	15500	15500
Min. Value	54.7	41.7	53.9	78	70.2
Aver. Value	5360	5350	5350	5350	5350
Energy	2.3	2.3	2.3	2.3	2.3
Interval Time	468.95	467	469.8	470	468.4

Remarks

Scenario E - Data sent by the motherboard (data sending power -> Normal Message)

Location 1	Average
WUT Time	77.27 ms
WUT Energy	1.078 µWh
DATA_TX Time	662.62 ms
DATA_TX Aver. Cur	47400 µA
DATA_RX Time	2257.82 ms
DATA_RX Energy	39.1 µWh

<u>Remarks :</u>

We are supposed to have DATA_TX Time = 413.7 ms (airtime LoRa calculator/Spreading Factor = 11)
Δ = 248.92 ms

Measure	1	2	3	4	5
WUT_ST Time	78.07	78.25	76.4	76.95	76.7
WUT_ST Energy	1.08	1.1	1.07	1.07	1.07
DATA_TX Time	661.98	663.2	663	662	662.9
DATA_TX Aver. Cur	47400	47400	47500	47200	47500
DATA_RX Time	2258.3	2256.9	2258.4	2257.9	2257.6
DATA_RX Energy	39.1	39.1	39.1	39.1	39.1

Configuration 2.4 : Sound board with accumulation data, polling and thresholds (ID 10)

Scenario A - Nothing happens (static power / sleep mode)

Location 1	Average
Maximum Value	385 µA
Minimum Value	49.2 µA
Average Value	50.38 µA

Measure	1	2	3	4	5
Max. Value	449	448	218	521	289
Min. Value	49.2	49.2	49.2	49.2	49.2
Aver. Value	50.4	50.4	50.3	50.4	50.4

Remarks :

CAUTION ! *When an exceeded threshold event occurs, the microphone is switched OFF. The average current consumption switches from 50.38 µA to 28.9 µA during 64000 ms.*

So, to obtain the static value of the sound board alone, we can do 50,38 µA -27,89 µA and we obtain ***22,49 µA*** *for the static current of the sound board alone. (27,89 µA is the static current of the motherboard used for this test)*

Scenario B - Threshold event (not exceeded) (dynamic power 1)

Location 1	Average
Maximum Value	6982 µA
Minimum Value	-22.15 µA
Average Value	4058 µA
Energy	1.734 µWh
Interval Time	466.2 ms

Measure	1	2	3	4	5
Max. Value	6990	6930	6960	7030	6930
Min. Value	-50.8	-42	-19.9	-8.23	10.2
Aver. Value	4060	4060	4040	4070	4060
Energy	1.73	1.73	1.74	1.74	1.73
Interval Time	465.3	466.2	469.3	465.9	464.4

Remarks

Scenario C - Threshold event (exceeded) (dynamic power 2)

Location 1	Average
Maximum Value	6982 µA
Minimum Value	63.62 µA
Average Value	4128 µA
Energy	1.682 µWh
Interval Time	445.24 ms

Measure	1	2	3	4	5
Max. Value	6930	6990	6950	7100	6940
Min. Value	50.4	85.5	61.8	70.5	49.9
Aver. Value	4110	4130	4130	4130	4140
Energy	1.68	1.69	1.67	1.68	1.69
Interval Time	446.4	446.4	442.8	444.9	445.7

Remarks

Scenario D - Polling interrupt (dynamic power 3)

Location 1	Average
Maximum Value	15600 µA
Minimum Value	85.68 µA
Average Value	5356 µA
Energy	2.29 µWh
Interval Time	466.98 ms

Measure	1	2	3	4	5
Max. Value	15600	15700	15600	15600	15500
Min. Value	66.7	112	102	73.3	74.4
Aver. Value	5360	5350	5360	5360	5350
Energy	2.29	2.3	2.29	2.29	2.29
Interval Time	465.8	467	467.4	467.3	467.4

Remarks

Special Measurements

Special Measurement 4.1 - Accumulated message save (for one metric)

Location 1	Average
Maximum Value	17540 µA
Minimum Value	-276.2 µA
Average Value	9626 µA
Energy	0.38 µWh
Interval Time	43.1 ms

Measure	1	2	3	4	5
Max. Value	17500	17600	17600	17500	17500
Min. Value	-288	-267	-278	-260	-288
Aver. Value	9670	9600	9650	9550	9660
Energy	0.383	0.377	0.379	0.378	0.383
Interval Time	43.2	42.9	42.9	43.2	43.3

Remarks

CAUTION! *A data saving from an exceeded threshold provokes this kind of power consumption. A data saving from a polling interrupt provokes a power consumption multiplied by 2 (like a power consumption for a two-metric data saving)*

Special Measurement 4.2 - Data sent by the motherboard (accumulated message)

Location 1	Average
WUT Time	133.94 ms
WUT Energy	1.906 µWh
DATA_TX Time	1236.96 ms
DATA_TX Aver. Cur	47120 µA
DATA_RX Time	2255.72 ms
DATA_RX Energy	39.3 µWh

Remarks :

We are supposed to have DATA_TX Time = 905.22 ms (airtime LoRa calculator/Spreading Factor = 11)
-> Δ = 331.74 ms

Measure	1	2	3	4	5
WUT_ST Time	133.2	134	134.7	134	133.8
WUT_ST Energy	1.9	1.91	1.92	1.9	1.9
DATA_TX Time	1237.4	1237.2	1235.9	1237	1237.3
DATA_TX Aver. Cur	47300	47300	45500	47800	47700
DATA_RX Time	2253.7	2252.4	2258.7	2257.4	2256.4
DATA_RX Energy	39.2	39.1	40	39.1	39.1

4.4 Conclusion

At the end of the tests, **no major differences were noticed between ID7 and ID9**. Therefore, we can conclude that the enabling/disabling of the polling feature has no influence on the static current. However, when we enable/disable the threshold feature, we can notice a major difference for the static current. For the total estimation, we need to create two sections : “Sound with no thresholds” (ID7 and ID9) and “Sound with thresholds” (ID8 and ID10).

For the total estimation, it’s not valuable to keep these four configurations so **we keep the ID9 (which becomes ID 4 in the excel file) and the ID10 (which becomes ID 5 in the excel file)**.

We notice the effect of disabling the microphone during one minute after an exceeded threshold occurs. This effect must be taken into account for the total estimation of the consumption. Also, after the special measurement 4.1, we notice a strange effect when data are saved in the motherboard. Indeed, a data saving from an exceeded threshold has a normal behaviour but it’s not the same with a data saving from a polling interrupt. It provokes the same behaviour than a saving operation for two metrics. Again, we need to take into account this strange effect in the total estimation.

Date of measurements : 21/01/2021

5. The environmental board

5.1 Description

5.2 Hardware analysis

5.3 Measurements

Modus operandi

1. Plug the motherboard, the environmental board and the *Otii ARC* according to the scheme X.X (refer to annex A in order to understand how to configure the measuring device *Otii ARC*).

2. Set the system configurations via the *IWAST configurator app* (refer to annex B in order to understand how to use this application). Several configurations must be set:

   2.1 **Environmental board without accumulation data, without polling and without thresholds (ID 11)**:
   - The accumulation data feature must be disabled
   - The polling interrupts must be disabled
   - The thresholds (high and low) must be disabled

   2.2 **Environmental board without accumulation data, without polling but with thresholds (ID 12)**:
   - The accumulation data feature must be disabled
   - The polling interrupts must be disabled
   - The thresholds (high and low) must be enabled
   - The threshold high for the metric 1 must be equal to XX
   - The threshold low for the metric 1 must be equal to XX
   - The threshold high for the metric 2 must be equal to XX
   - The threshold low for the metric 2 must be equal to XX
   - The threshold high for the metric 3 must be equal to XX
   - The threshold low for the metric 3 must be equal to XX
   - The threshold high for the metric 4 must be equal to XX
   - The threshold low for the metric 4 must be equal to XX

   2.3 **Environmental board without accumulation data, without thresholds but with polling (ID 13):**
   - The accumulation data feature must be disabled
   - The polling interrupts must be enabled and set to 2 min
   - The thresholds (high and low) must be disabled

   2.4 **Environmental board with accumulation data, polling and thresholds (ID 14):**
   - The accumulation data feature must be enabled
   - The polling interrupts must be enabled and set to 2 min
   - The thresholds (high and low) must be enabled
   - The threshold high for the metric 1 must be equal to XX
   - The threshold low for the metric 1 must be equal to XX
   - The threshold high for the metric 2 must be equal to XX
   - The threshold low for the metric 2 must be equal to XX
   - The threshold high for the metric 3 must be equal to XX
   - The threshold low for the metric 3 must be equal to XX
   - The threshold high for the metric 4 must be equal to XX
   - The threshold low for the metric 4 must be equal to XX

3. Collect the current measurements for each configuration at different moment. The test should last 15 min to be relevant. With this physical configuration, we can collect the values of the current flowing between the power supply and the motherboard (location 1). Multiple scenarios must be analysed with this module:

    A. When nothing happens (static power / sleep mode)
    B. When a threshold event (not exceeded) occurs (dynamic power 1)
    C. When a threshold event (exceeded) occurs (dynamic power 2)
    D. When a polling interrupt occurs (dynamic power 3)
    E. When data are sent by the motherboard (data sending power -> Normal Message)

Results

Configuration 2.1 : Environmental board without accumulation data, without polling and without thresholds (ID 11)

Configuration 2.2 : Environmental board without accumulation data, without polling but with thresholds(ID 12)

Configuration 2.3 : Environmental board without accumulation data, without thresholds but with polling (ID 13)

Configuration 2.4 : Environmental board with accumulation data, polling and thresholds (ID 14)

5.4Conclusion

Annex A - Configuration of the *OTII ARC*

The Otii Arc is a powerful tool designed to analyse the energy consumption of embedded systems. It was specifically developed to design energy efficient IoT products. Its use is very simple. First of all, you must download and install the associated software available for free on the website https://www.qoitech.com/download/. After that, follow the modus operandi bellow to collect measurements on the IWAST system :

1. Connect the Otii ARC to your computer and start a new project
2. Configure the new project with a power supply of 3.3 V
3. Connect the motherboard and the Otii ARC together
4. Enable the power supply via the application and start a new recording

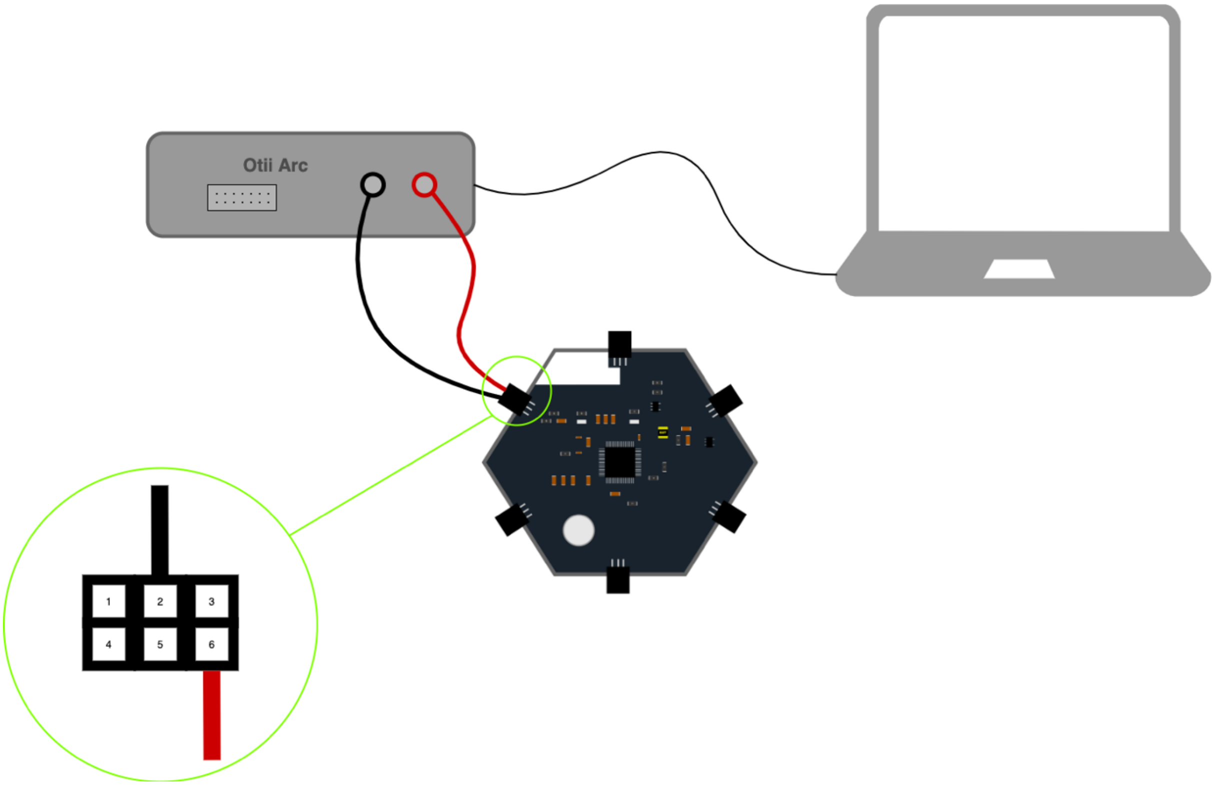

Figure 28 Configure and connect the Otii Arc

Pins Motherboard	Description
3	External interrupt (for polling)
2	Ground
1	Sensor Power 3.3 V
6	Power Supply of the system
5	SCL pin (I2C)
4	SDA pin (I2C)

Table 8 Pin description of the motherboard

Annex B - Using the *IWAST Configurator*

The *IWAST Configurator* is the application made by *DRAMCO* to easily configure the *IWAST* system. First of all, you must download the installation file accessible directly on the GitHub page of IWAST https://github.com/dramco-iwast/qt-config/tree/master/target (choose the last version if possible). The installation is very easy and does not require too much storage space. When the app is correctly installed, you can start a configuration by following this modus operandi :

1. Open the application.
2. Connect the IWAST system to your computer with the USB cable. Make sure that all sensor boards you wish to use are connected to the motherboard before connecting the system to your computer.
3. Click on the “Refresh” button until “Arduino Zero” appears in the drop-down list of COM Ports.
4. Select the “Arduino Zero” port in the list and click on the “Connect” button.
5. Configure the sensor boards by clicking on the “Load” button of each sensor board. When you have chosen all the parameters you want for a sensor board, click on the “Save” button. Repeat this step for each sensor boards.
6. When the sensor board configurations are finished, click on the “Disconnect” button.
7. You can start again a complete configuration for a new IWAST system by clicking on the “New” button. In this case, go back to the step 2.

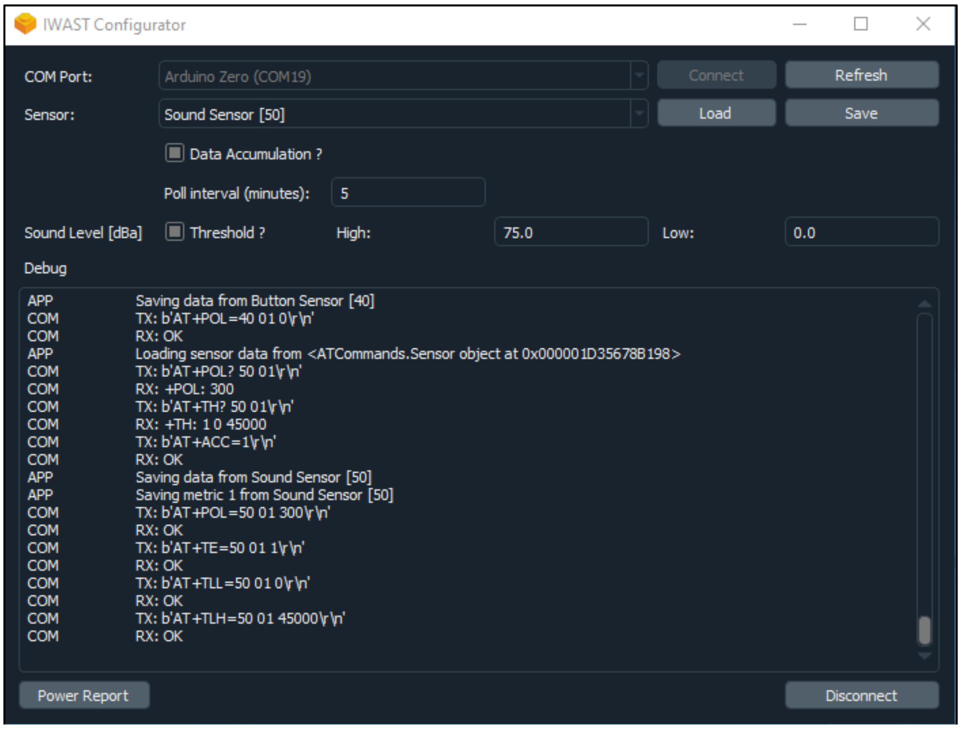

Figure 29 Configuration example